Fall 10-24-2012

A GPU-Based Real- Time Event Detection Framework for Power System Frequency Data Streams

Olufemi Omitaomu
Oak Ridge National Laboratory

Kyle Spafford
Oak Ridge National Laboratory

Steve Fernandez
Oak Ridge National Laboratory

A GPU-Based Real-Time Event Detection Framework for Power System Frequency Data Streams

Olufemi Omitaomu

Kyle Spafford

Steve Fernandez

October 24, 2012

U.S. DEPARTMENT OF ENERGY

Modeling, Simulation and Optimization for the 21st Century Electric Power Grid

Lake Geneva, Wisconsin, USA

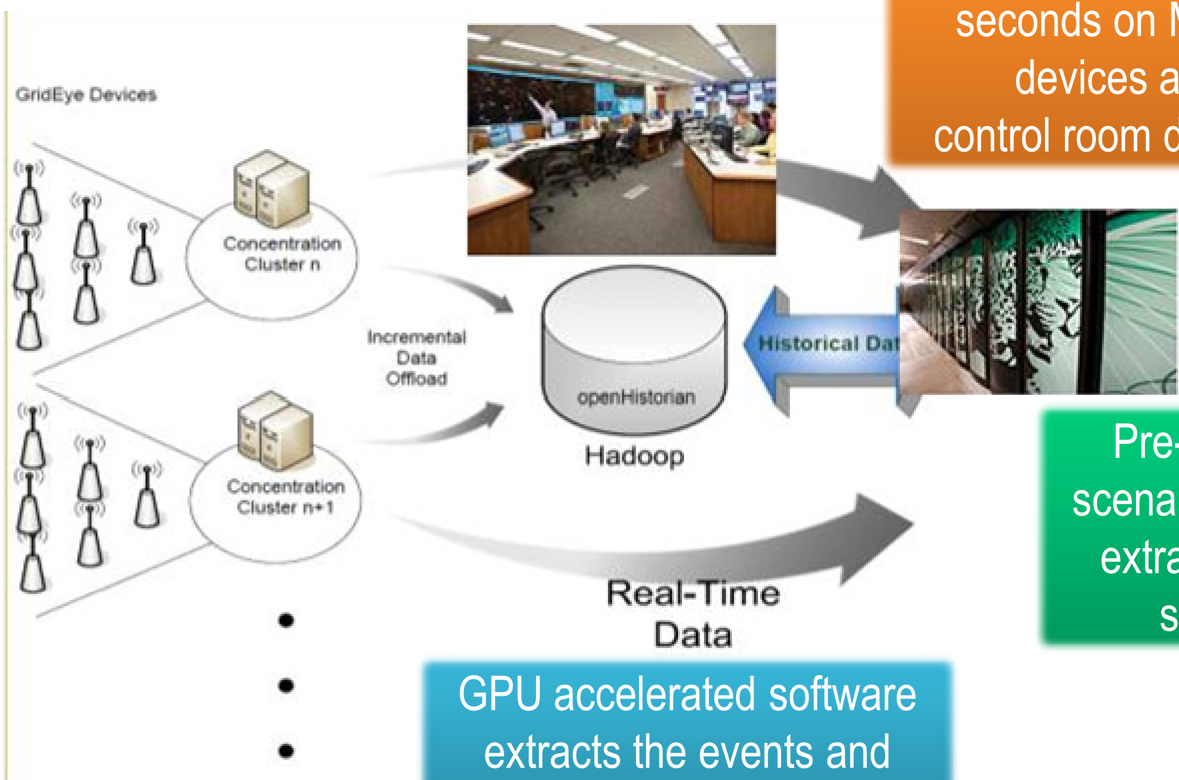
October 21 – 25, 2012

OAK RIDGE NATIONAL LABORATORY

MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Overview

Detectors sense event within one second of event



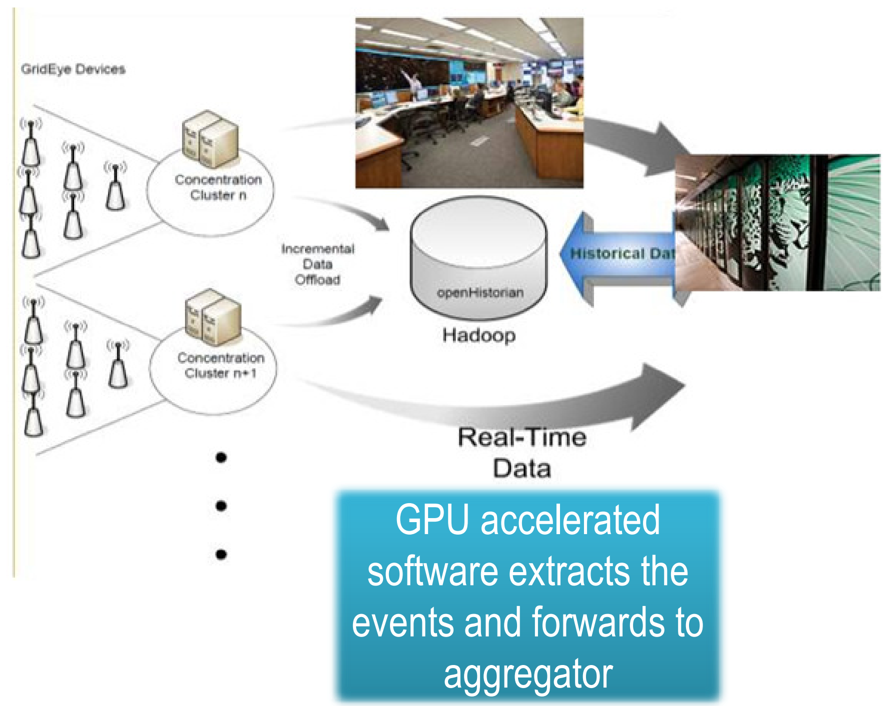
Display to decision makers within 2-4 seconds on Mobile devices and control room displays

Pre-calculated 10^8 scenarios with patterns extracted from HPC stored library

GPU accelerated software extracts the events and forwards to aggregator

Advanced Event Extraction Algorithms

- Imbalance of generation and load can cause sudden frequency changes in the system
- Some events of concern:
  - Generator or Line Trips
  - Load Rejection
  - Oscillations
- Two real-time goals:
  - Detecting the occurrence of events using sensor data
  - Identifying root cause using simulations



Proposed Applications

- Real time dynamic modeling
- Visualization and situational awareness
- High resolution state estimation/observation
- Model improvement and validation
- Load assessment and forecasting (renewable and DG integration)
- Advanced relaying and other protective schemes
- Advanced closed loop control systems

What are the Challenges?

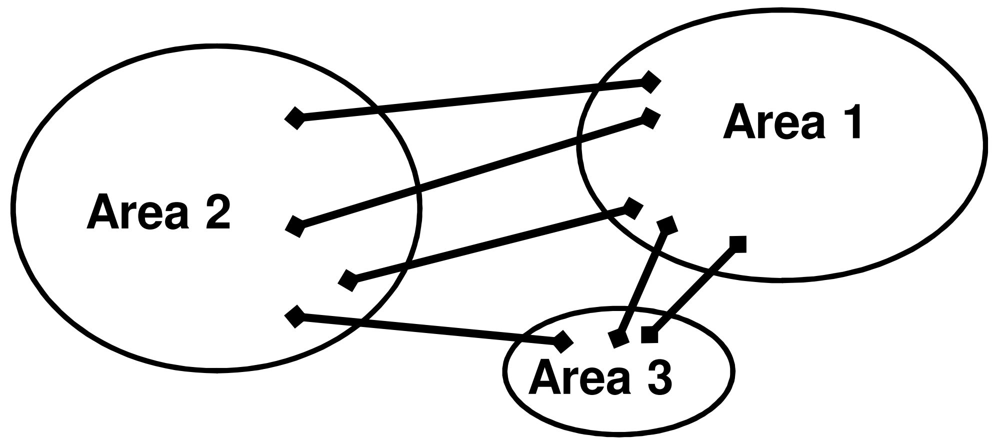
- Real time processing of sensor data streams (1-2 TB/hour)
- Archive of at least one year of sensor data (8.67-17.52 PB)
- Identify impending disruptions faster enough and within operators' decision loop
- Large batches of data-intensive simulations
- Noisy data, missing values
- Need single pass event detection algorithms with small memory requirements
- Streaming data analytics

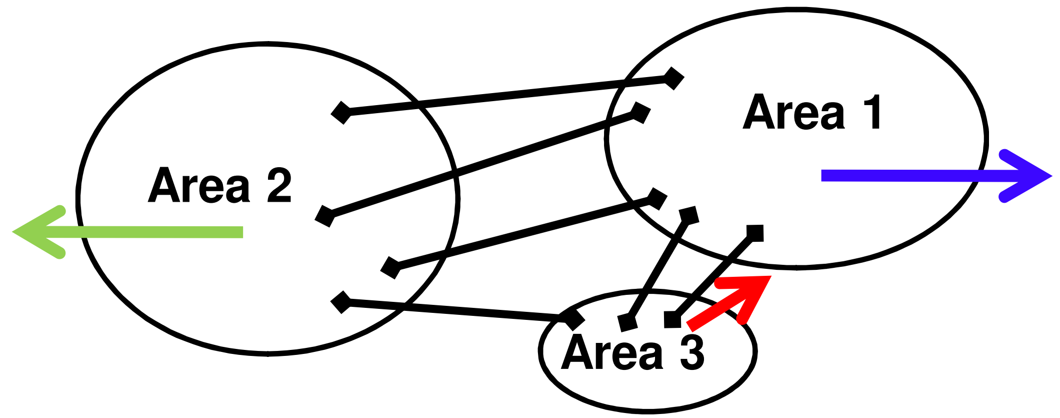
Inter-area Oscillations

- Oscillations associated with groups of generators
- Frequencies in the range of 0.1 to 0.8 Hz
- Factors influencing these modes are not fully understood
- If left unchecked, it could lead to cascading blackouts
- Our Approach explores a nonlinear and non-stationary technique for extracting inter-area modes

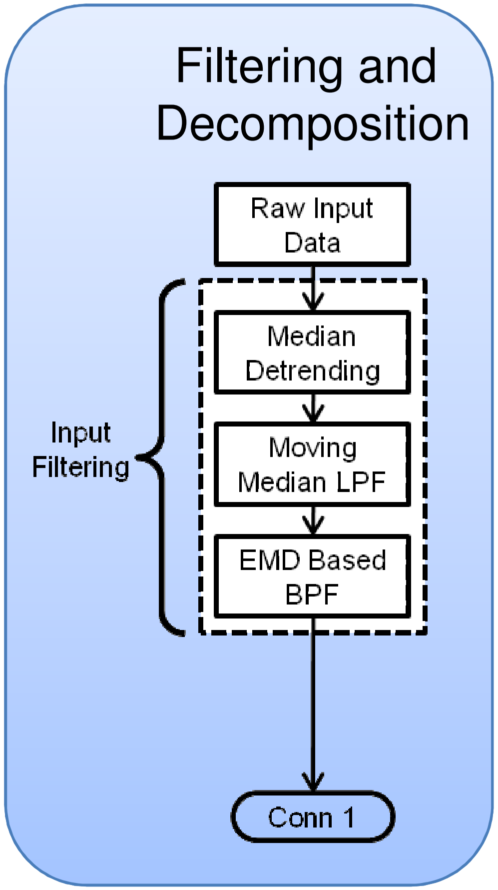
Inter-area Oscillations

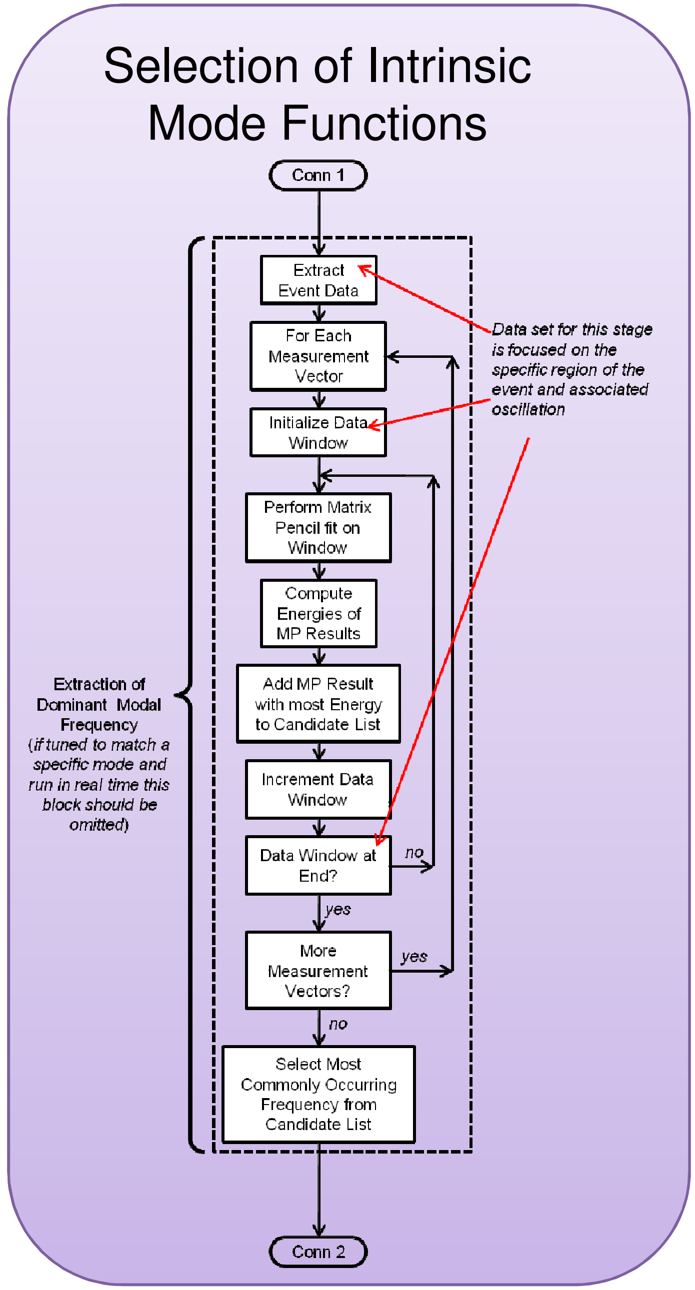
- The basics of the procedure will be demonstrated by this simple system
- An event incites an inter-area oscillation
- Area 1 is oscillating against Area 2
- Area 3 closely agrees with Area 1





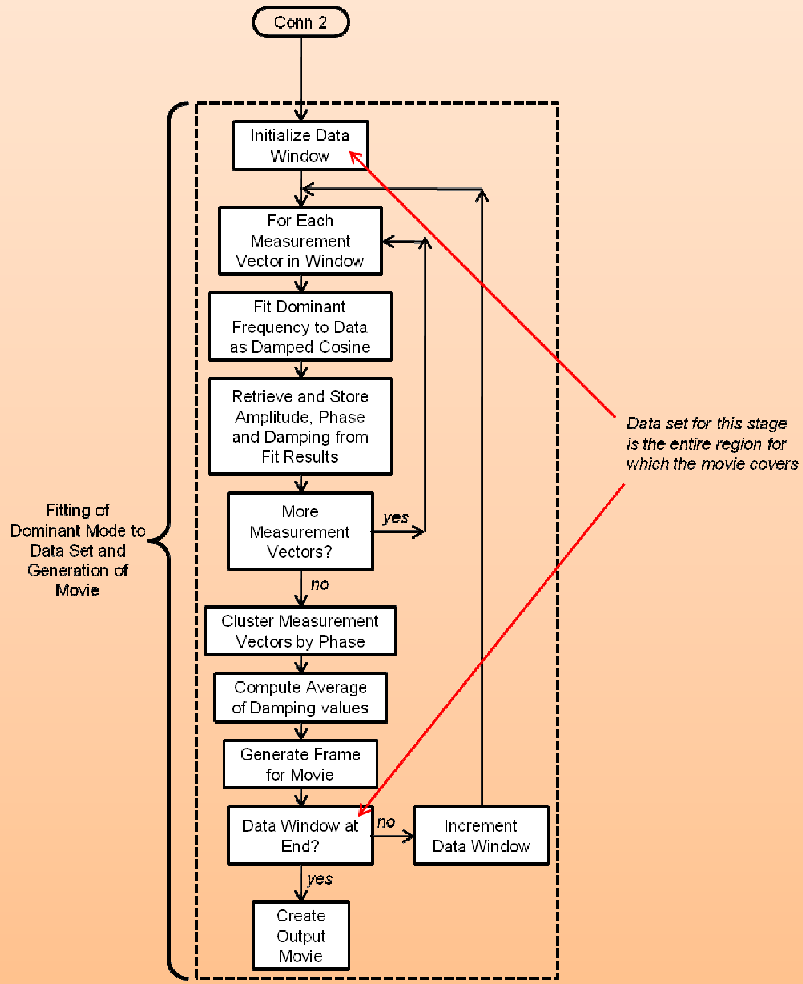
Detecting Inter-area Oscillations





Visualizing Inter-area Oscillations

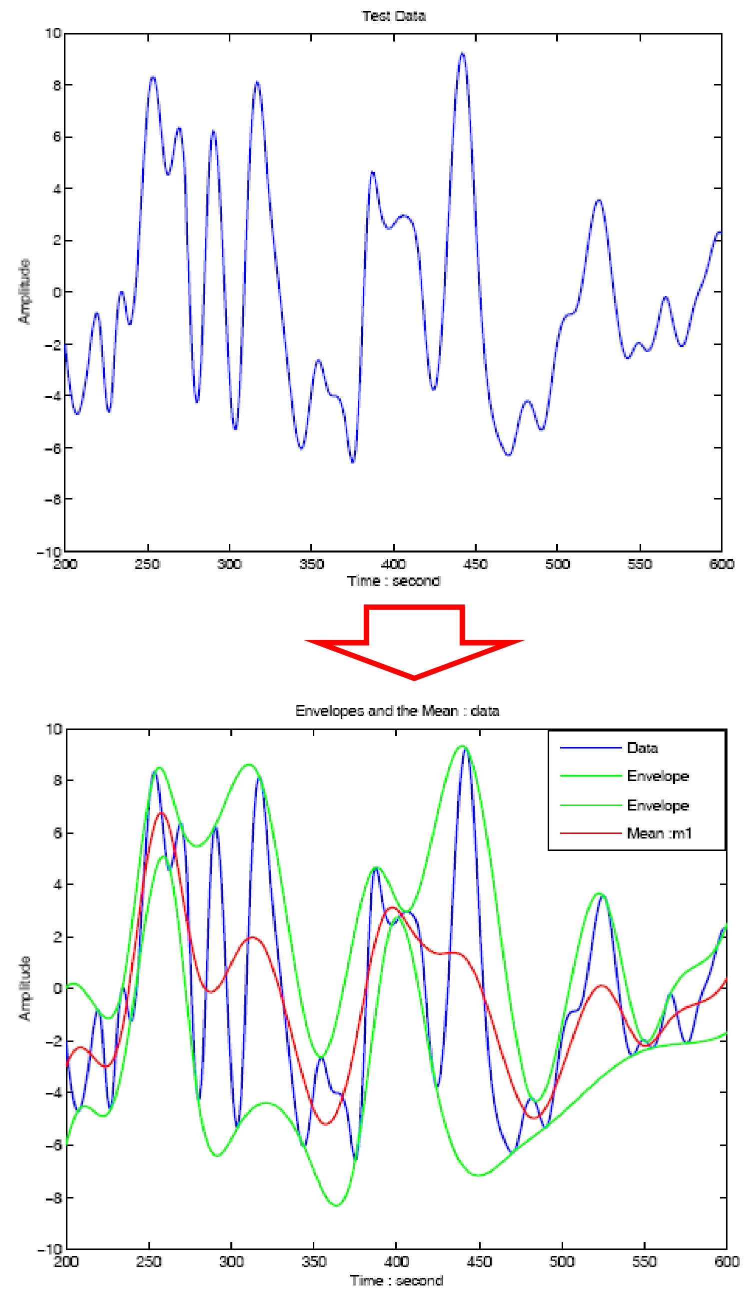
Clustering of Modes and Visualization



Empirical Mode Decomposition (EMD)

- Consider a 1-D signal, x_j sampled at times $t_j, j = 1, \ldots, N$
- Identification of the maxima and minima of the signal
- Interpolation of the set of maximal and minimal points
- Calculate the point-by-point average of the upper and lower envelopes

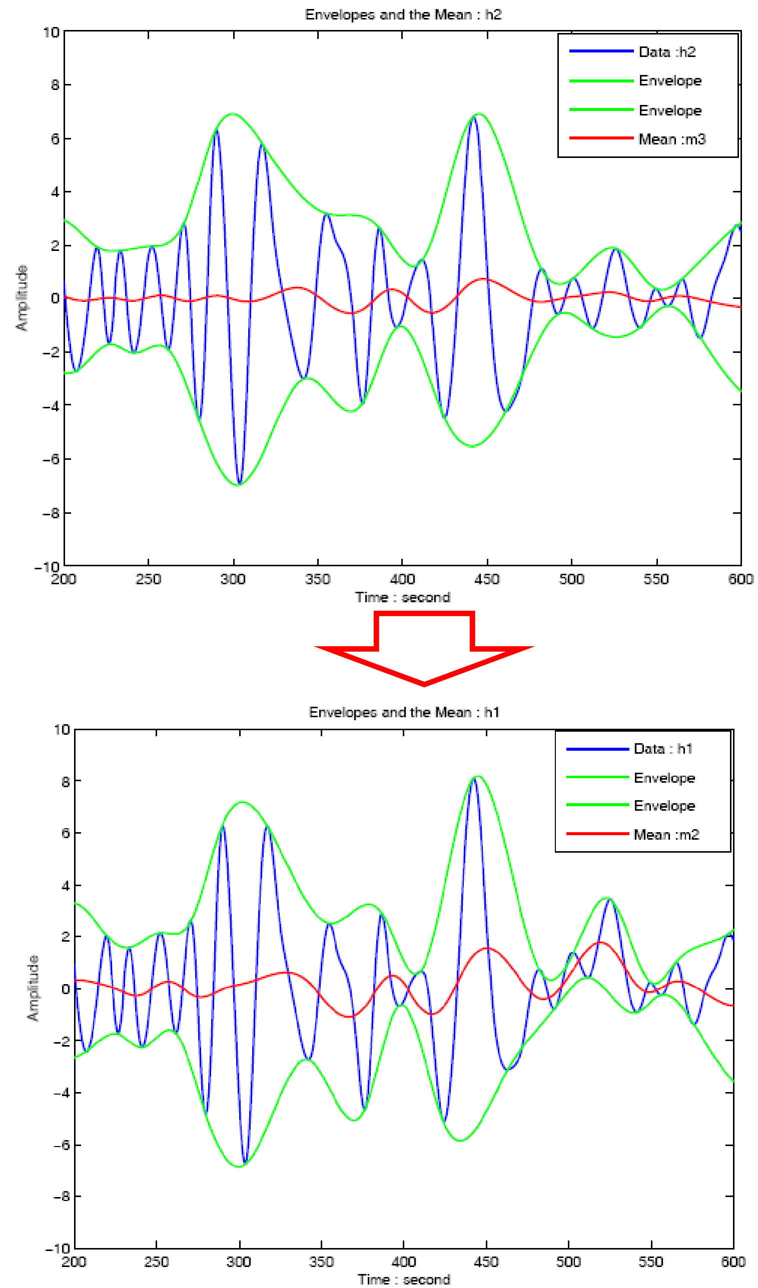
$$m_j = \left(x_{j_{up}} + x_{j_{low}} \right) / 2$$



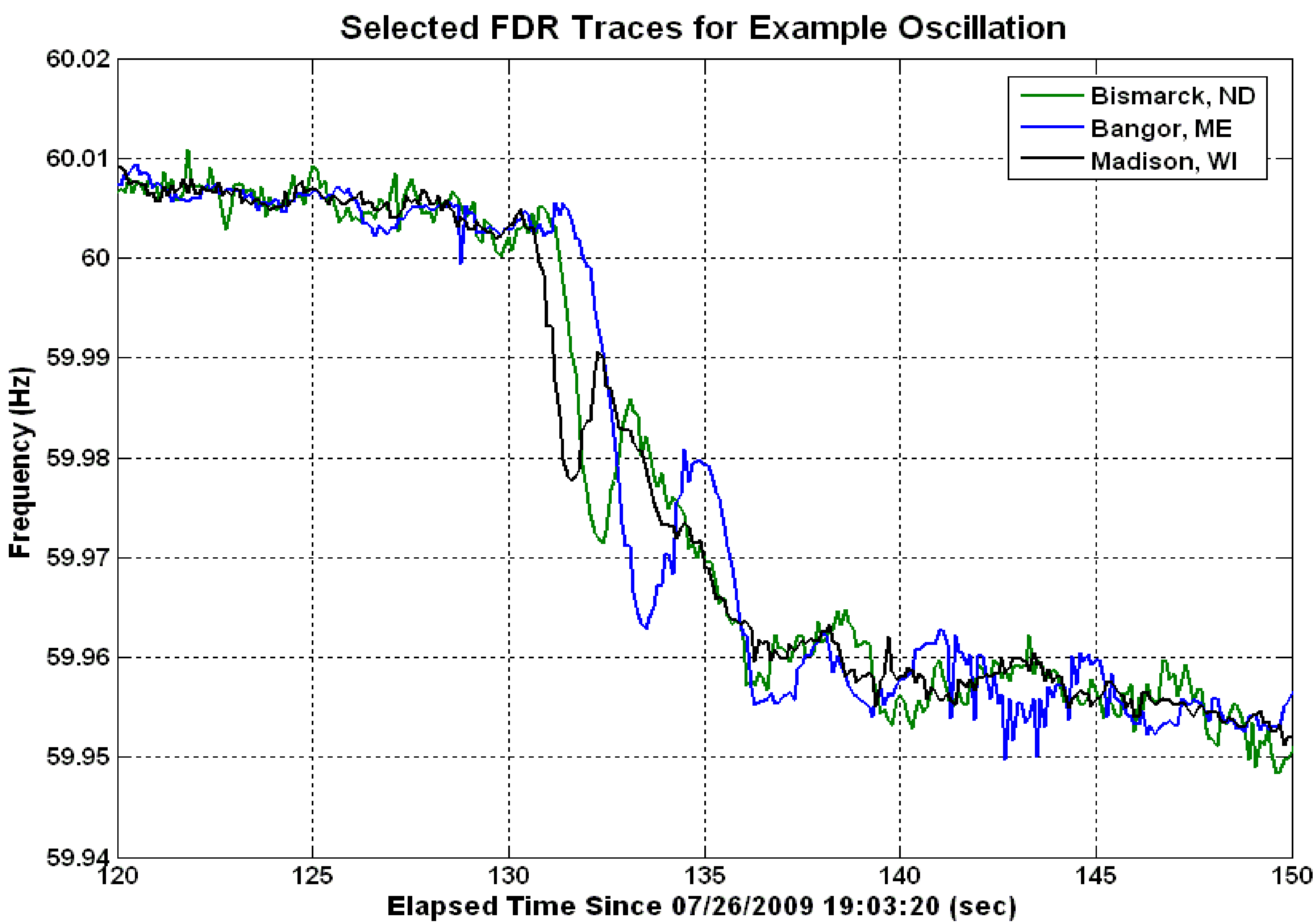
Empirical Mode Decomposition (EMD)

- Subtract the average from the original signal $d_j = x_j - m_j$
- If d_j is not an IMF, repeat the steps until d_j satisfies the two conditions for an IMF
- If an IMF is generated, the residual signal $r_j = x_j - d_j$ is regarded as the original signal and the steps are repeated for the 2nd IMF, and so on
- Finally,

$$x_j = \sum_{i=1}^{M-1} d_{j,i} + r_{j,M} \qquad i = 1, \dots, M$$

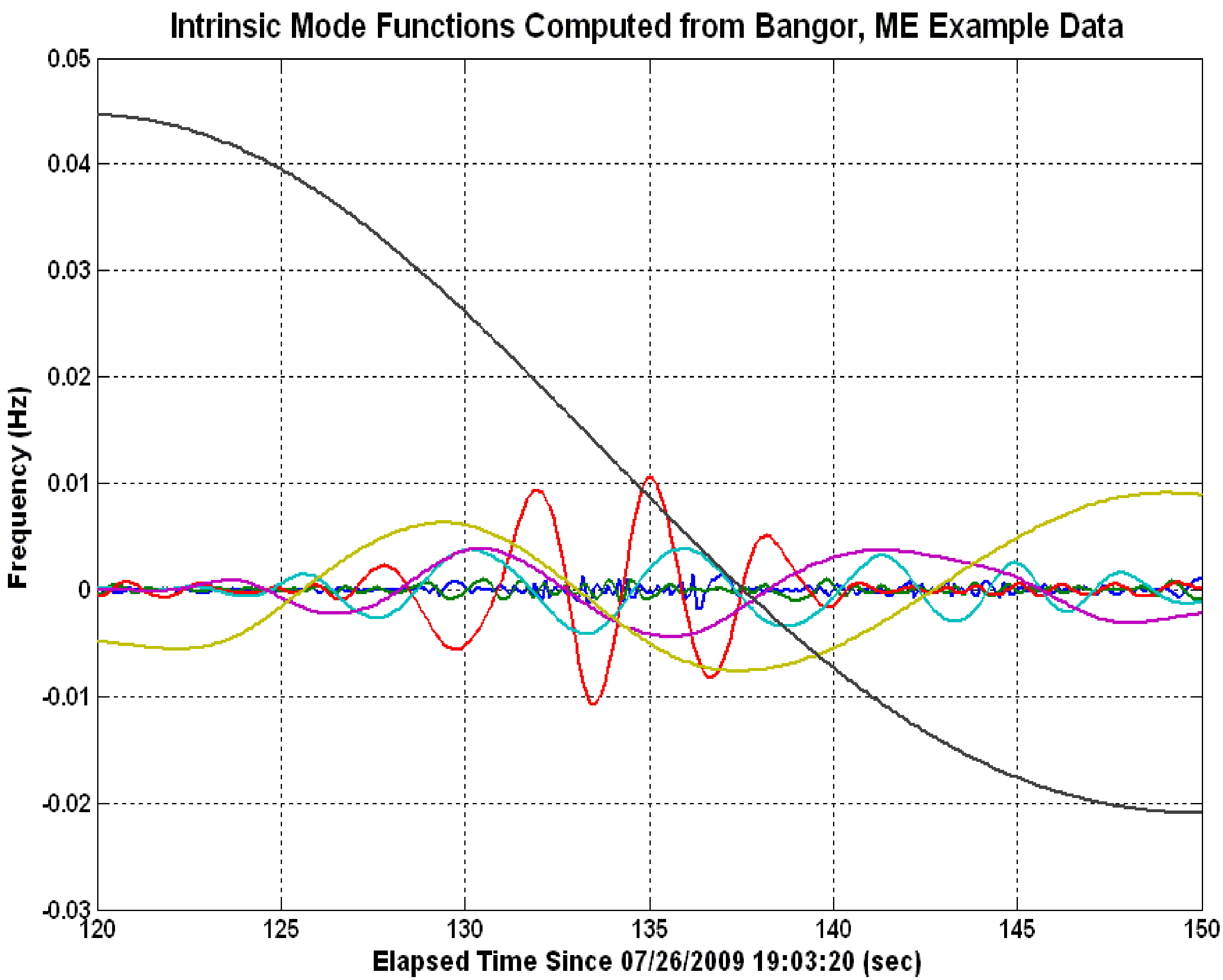


Sample Sensor Data

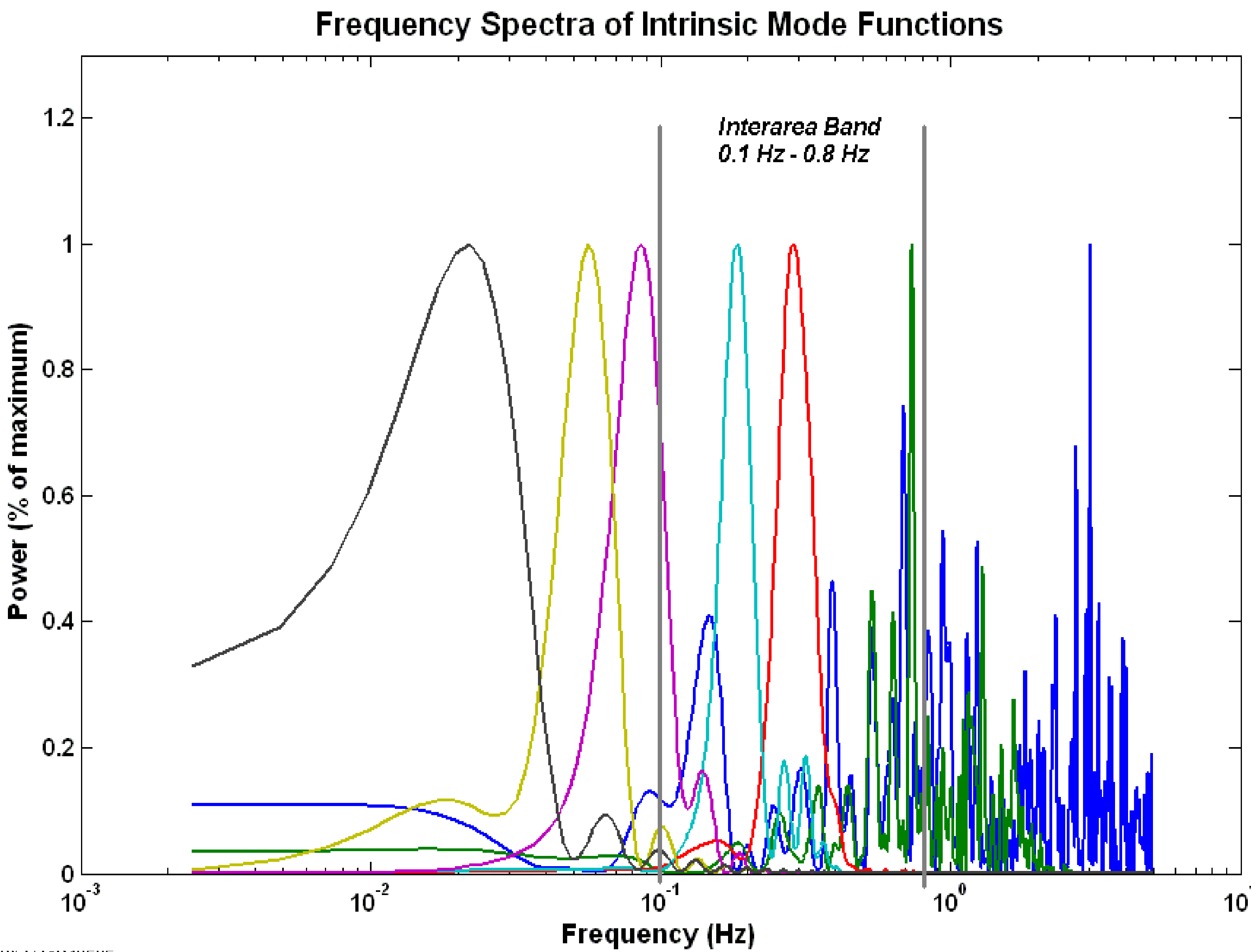


 Managed by UT-Battelle
for the U.S. Department of Energy

Decomposing the Signals using EMD



Clustering the IMFs using FFT



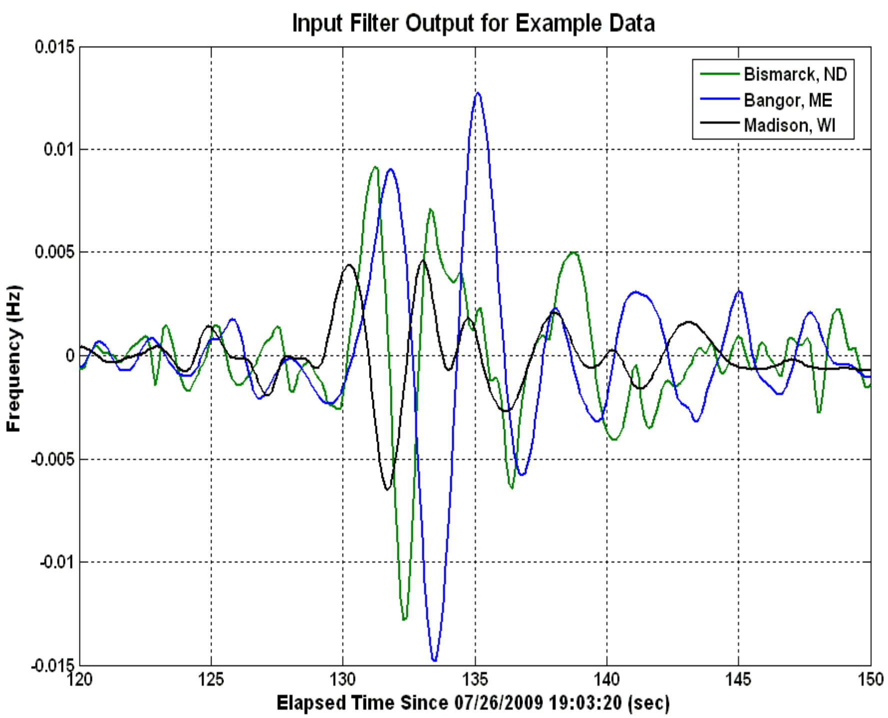
Selection on Inter-area IMFs

	Power In Band	Total Signal Power	Percent In Band (power threshold)	> 0.75
IMF 1	9.081E-06	1.522E-04	0.060	discard
IMF 2	9.796E-06	8.928E-05	0.110	discard
IMF 3	1.500E-03	1.600E-03	0.984	retain
IMF 4	4.540E-04	4.847E-04	0.937	retain
IMF 5	1.851E-04	3.500E-03	0.052	discard
IMF 6	4.300E-03	1.191E-01	0.036	discard

- The percentage of power within the interarea band is computed for each IMF

$$\%_{Interarea} = \frac{\sum_{f=0.1}^{0.8} P_f}{\sum_{f=0}^{f_{MAX}} P_f}$$

- This percentage is then compared to a set threshold to determine whether to retain the IMF



Fit Procedure

- An oscillation frequency, f_{OSC}, is predetermined by a Matrix Pencil based analysis
- The appropriate amplitude, phase and damping for this mode is established for each time step through a least squares fit to one oscillation period of the filtered signal

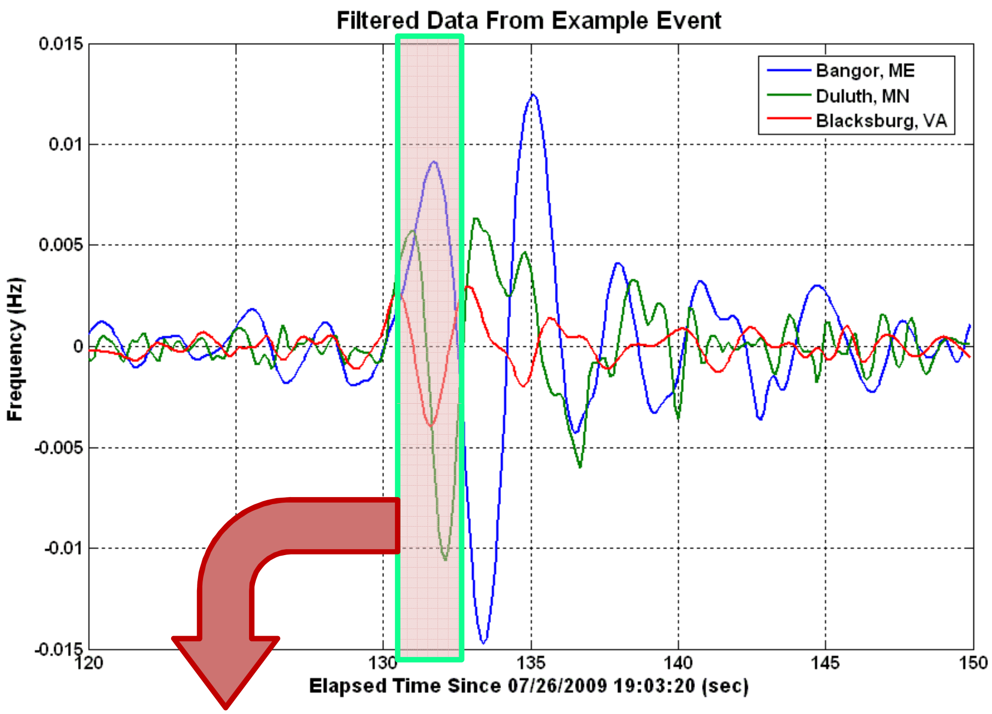
Modal Component => $y = Ae^{\alpha t}\cos(2\pi f_{OSC} t + \theta)$

Power => $P_y = y^2$

Total Energy => $E_y = \sum_t P_y$

A = Amplitude
α = damping
f = frequency
θ = phase angle
P and E = Analogous metrics

Damped Cosine Fit to Data Window



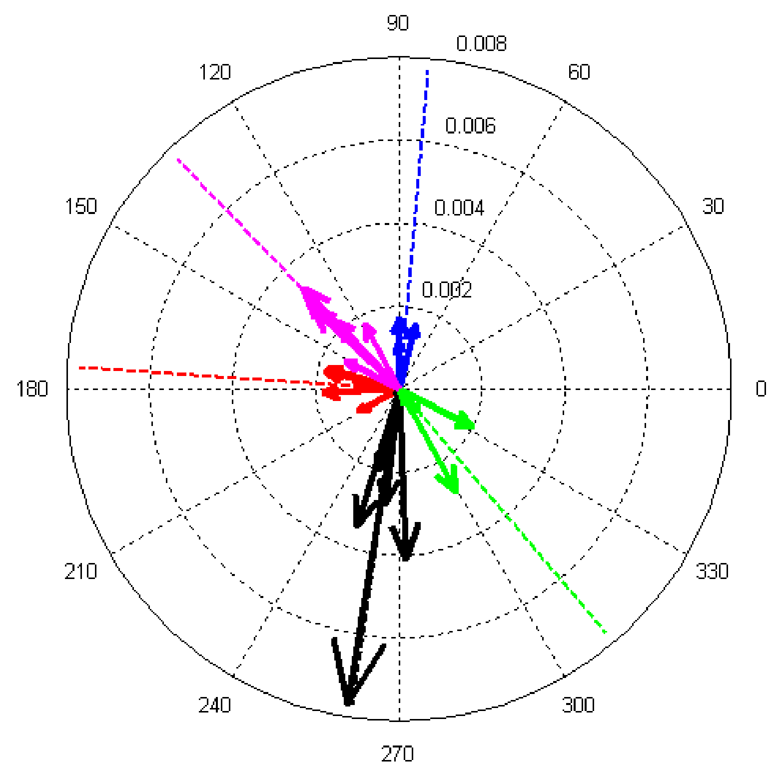
	Amplitude	Phase (deg)	Damping Factor
Bangor, ME	2.437E-03	153	0.454
Duluth, MN	6.935E-03	269	0.165
Blacksburg, VA	3.297E-03	335	6.499E-05

180^0 out of phase

- The damped cosine function is fitted to each measurement point within the data window
- The amplitude, phase and damping resulting from these fits are recorded
- The data window then moves to the next timestamp and the process is repeated

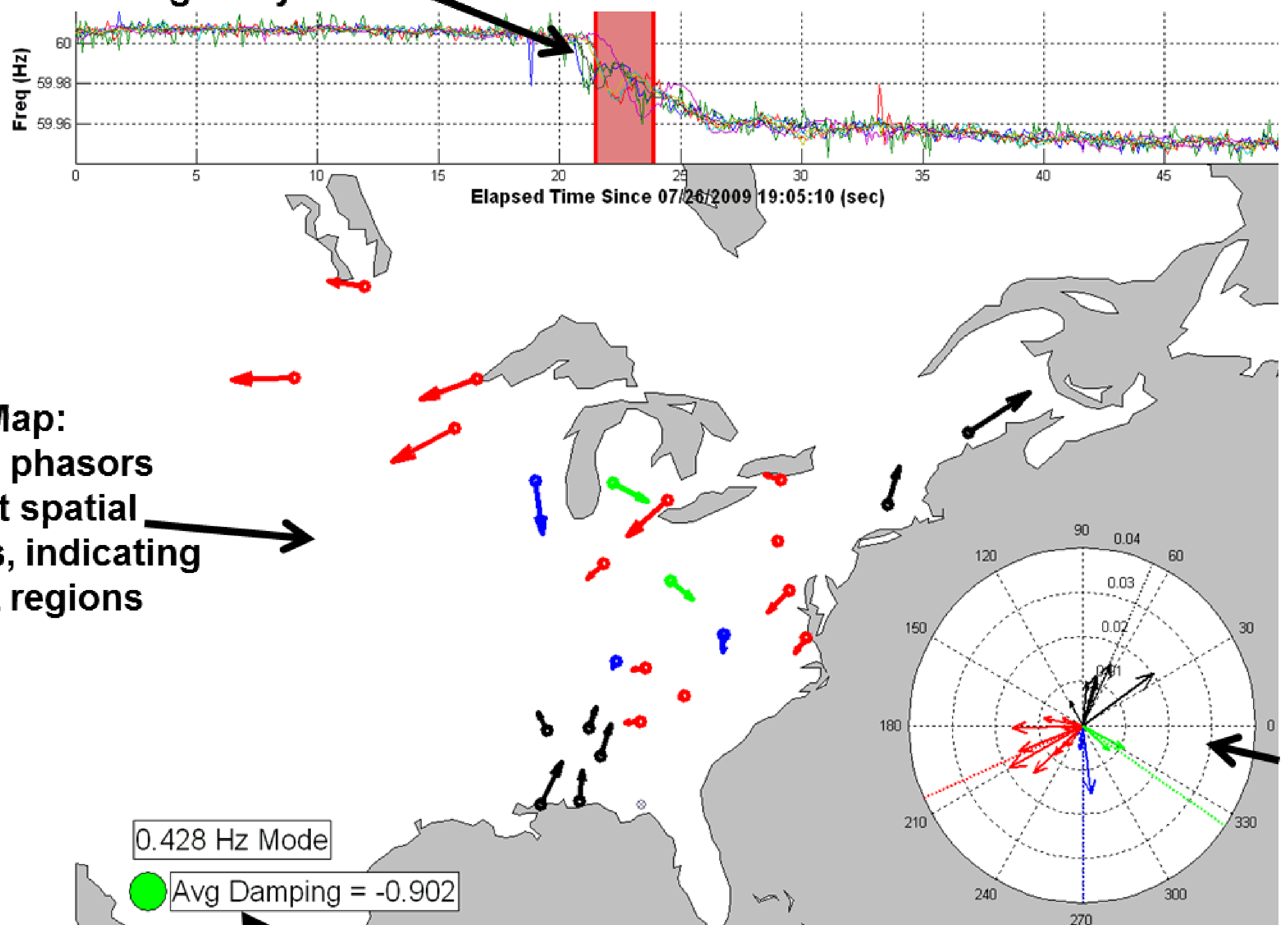
Identification of Coherent Groups

- With the phase angles determined for each time set the coherent groups are identified
- Achieved by clustering the mode phasors using phase angle
- In actuality the phasor projections are used to introduce dependence on the phasor amplitude (deweighting phasors with low amplitude)



Visualization of the Outputs

Time response of measurement data, gives the temporal location of data window being analyzed



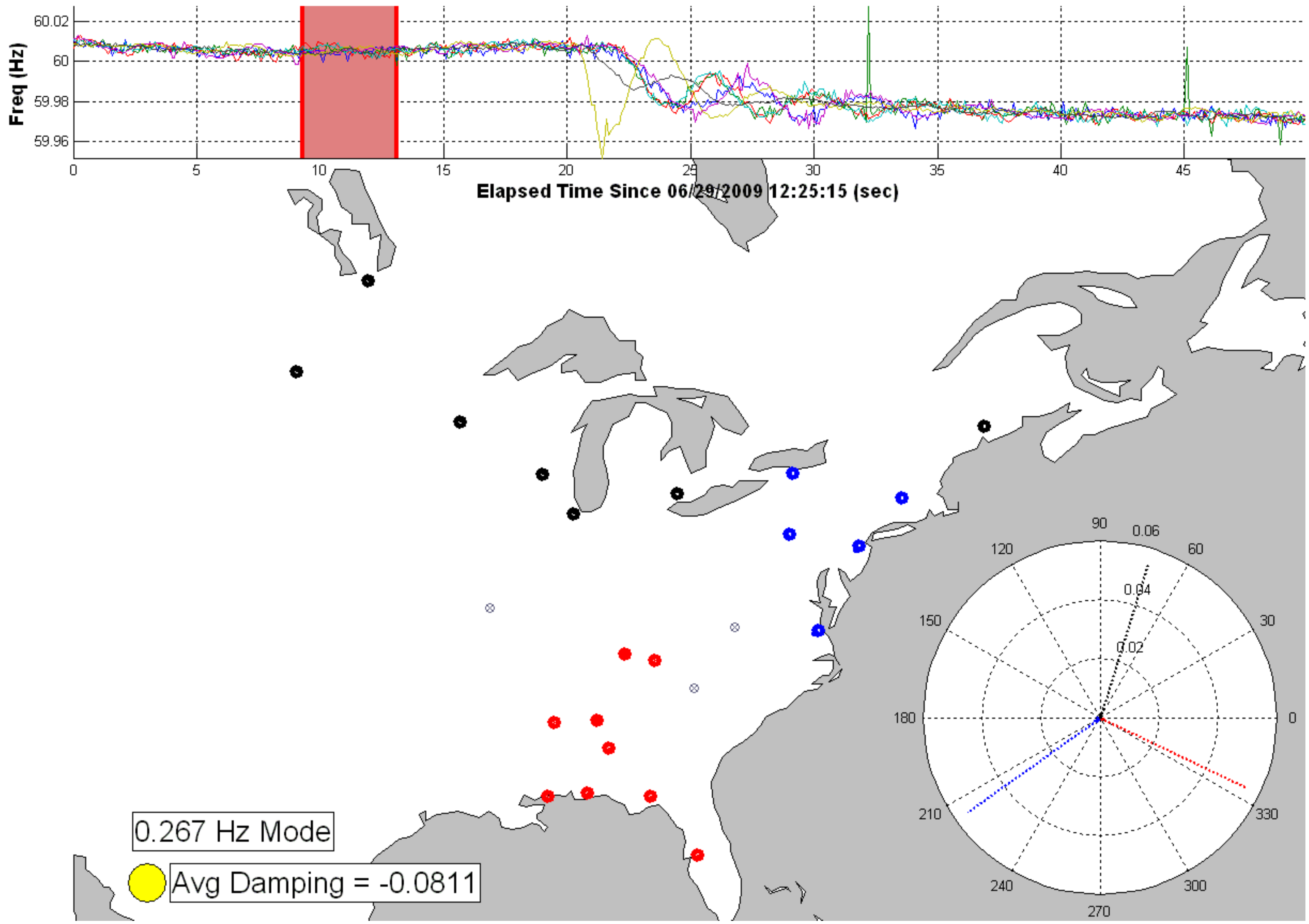
Phasor Map:
All mode phasors plotted at spatial locations, indicating coherent regions

Phasor Diagram:
All mode phasors plotted on same Axis to demonstrate groups and magnitudes

Mode Frequency, Average System Damping and Damping Alarm Light
Green if α < -0.1; Yellow if -0.1 < α < 0; Red if α > 0

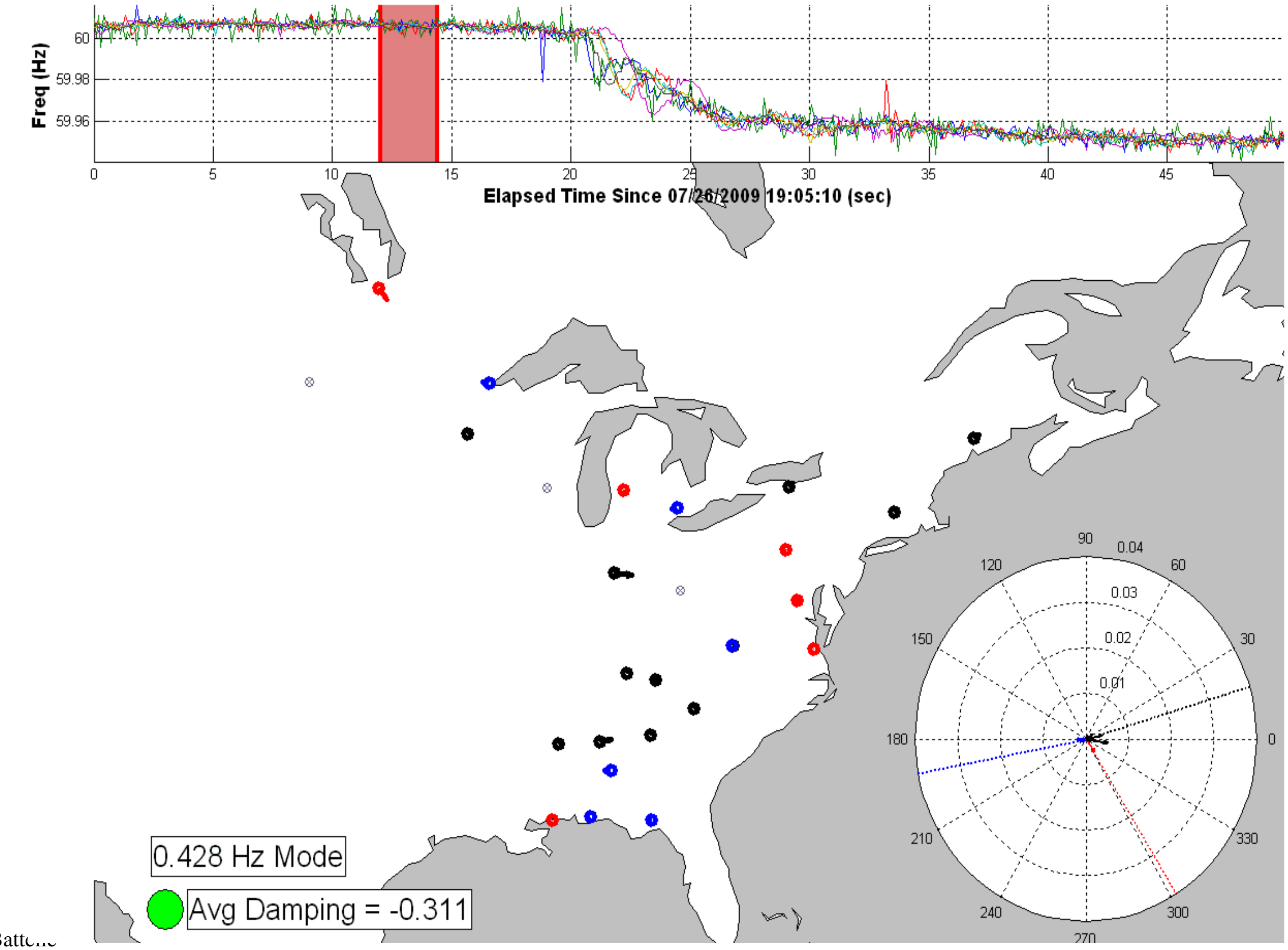
Case Study 1

- Loss of generator in Northeastern Florida – June 29, 2009



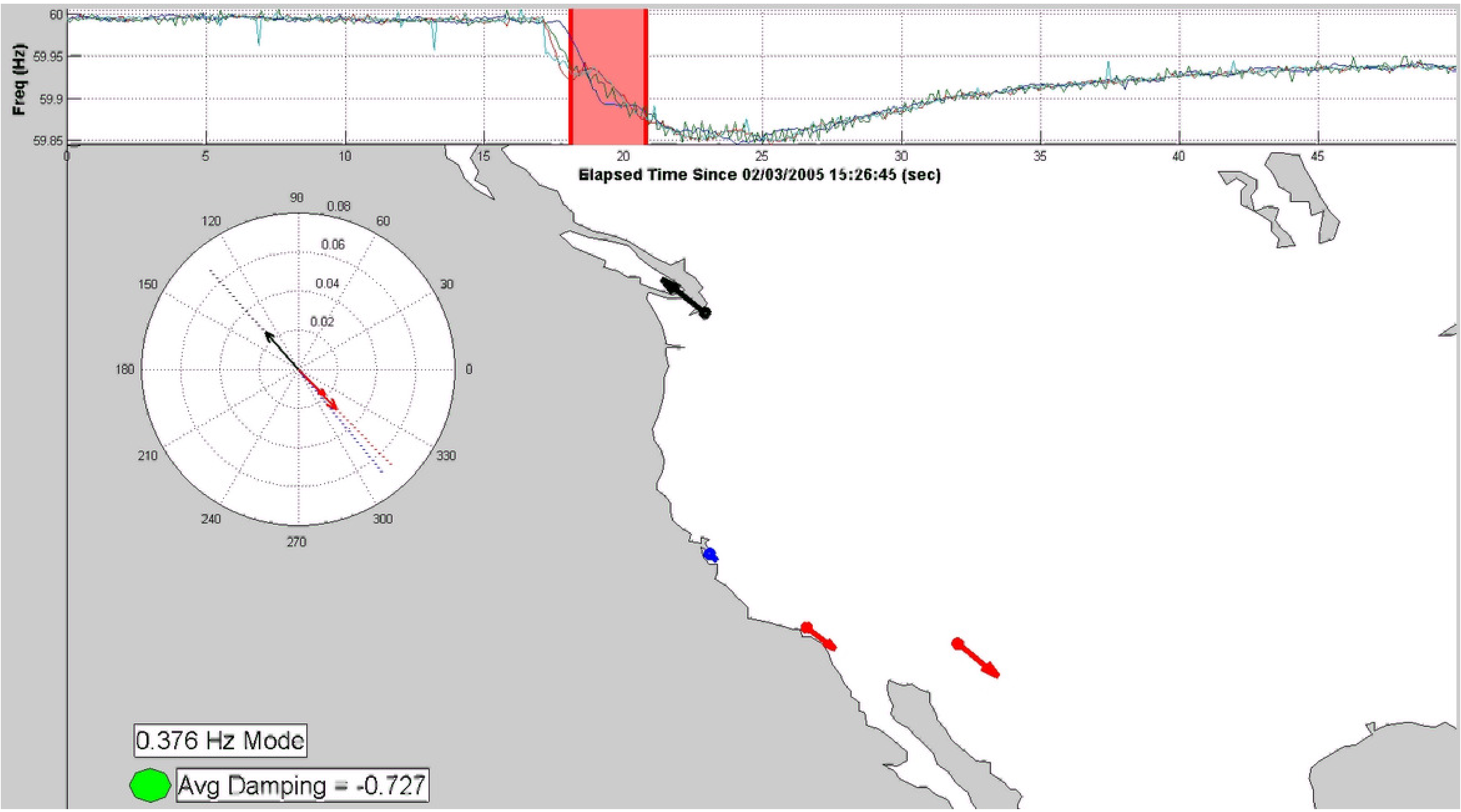
Case Study 2

- Generation trip at the Donald C. Cook nuclear power plant in Southwestern Michigan – July 26, 2009



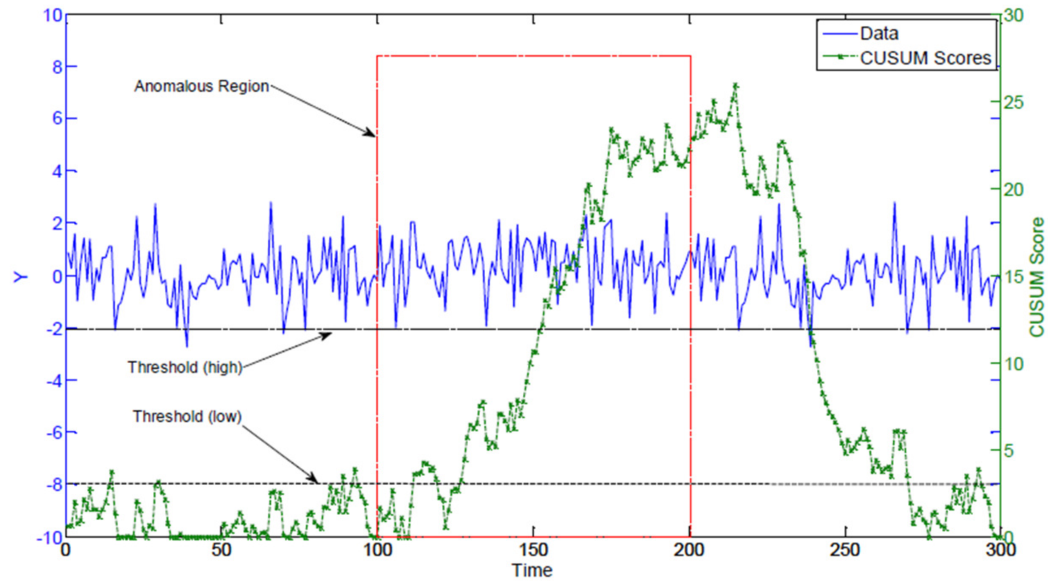
Case Study 3

- WECC Event



Frequency Change Detector using Cumulative Sum Control Chart

- Identify spans of the form $[t_1, t_2]$, such that the underlying system is in an anomalous state from time t_1 to t_2
- For a given time series, at a given time t (> 1), the following two quantities are computed:



$$S_t^+ = \max(0, S_{t-1}^+ + (T_t - \omega_t))$$

$$S_t^- = \max(0, S_{t-1}^- + (\omega_t - T_t))$$

Frequency Change Detector – Some Results

- Data used are from 21 single-phase PMUs within the Eastern Interconnect
- Data for two months – May and June 2008 – analyzed
- Data preprocessed using K-Median filter (k = 5)

Ignoring Spatial Information Has High False Positive Rate

FDR #	Location	State	May 2008		June 2008	
			Fraction Alarms	# Anomalous Events	Fraction Alarms	# Anomalous Events
11	Grand Rapids	MI	0.0082	117	0.0094	115
13	Carmel	IN	0.0081	115	0.0095	113
521	Chillicothe	OH	0.0076	117	0.0058	71
523	Oak Ridge	TN	0.0033	43	0.0019	19
597	Cookeville	TN	0.0067	92	0.0056	66

- Approximately 100-200 events detected per month
- Clearly a large false positive rate

Simple Spatial Smoothing Helps

FDR #	Location	State	May 2008	June 2008
11	Grand Rapids	MI	8	8
13	Carmel	IN	5	6
521	Chillicothe	OH	0	0
597	Cookeville	TN	0	0

- Spatial co-location constraint
  - An alarm is a “true event” if it is also raised at about the same time by neighboring sensors
- Significant reduction in false positive rate

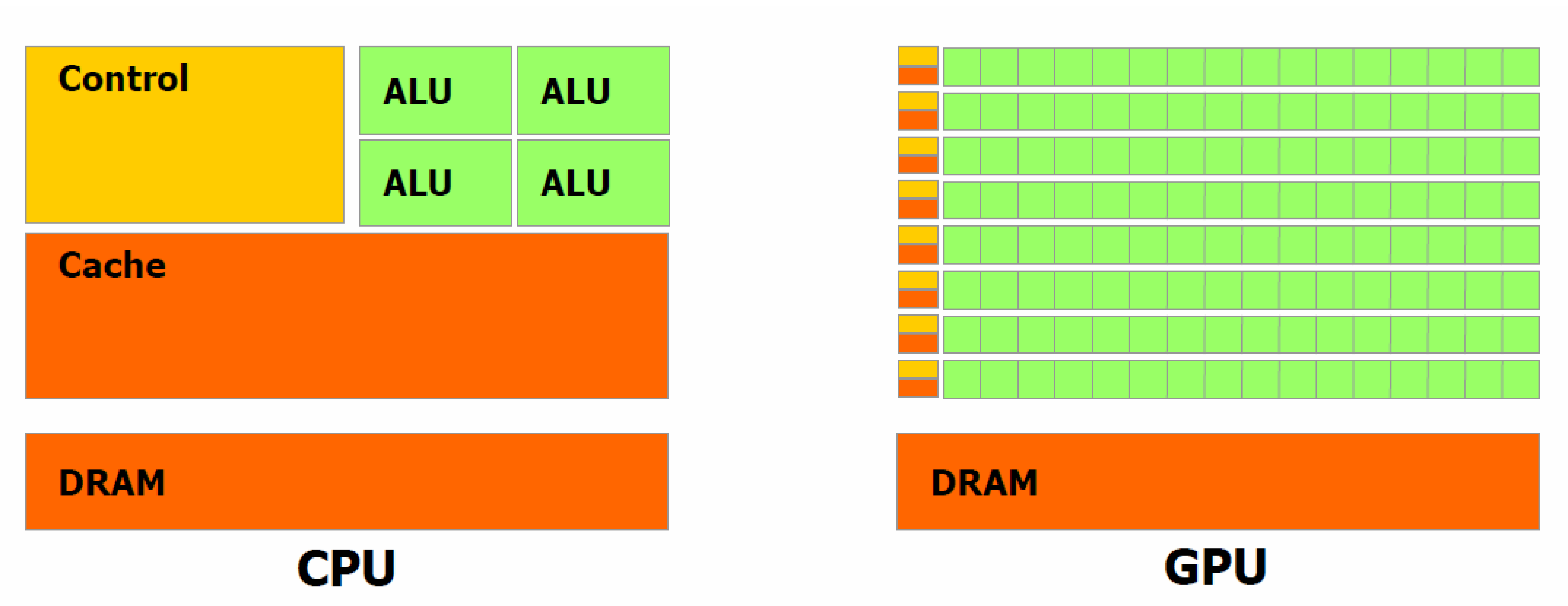
Performance Characteristics of Event Detection

- Based on a sliding window, so “working set” is small
- Highly dependent on:
  - Floating point performance
  - Memory bandwidth
- Extremely parallel
  - Every sensor’s data stream is independent

- To computer architects, this starts to sound extremely familiar…

It's close to the ideal case for a GPU!

- With less resources spent on cache, GPUs are more efficient for parallel problems with small working sets



GPU-Accelerated Compute Node

- Why GPUs?
  - Less resources on cache
  - Inexpensive
  - Energy Efficient
  - Horizontal Scaling (proportional increase in sensors and GPUs)
  - Fast compression of sensor data: 75 GB/s @ 1.25x compression ratio

How many GPU nodes will we need?

- Our experimental GPU clustering-based ED processes data at 1.2 GB/s
- An estimated 1-2 TB/hour total incoming data
  - Or 278-556 MB/s

- 1-2 GPUs are needed for initial ED processing
- Double for failover

Store and Analyze 8.7+ PB of Data

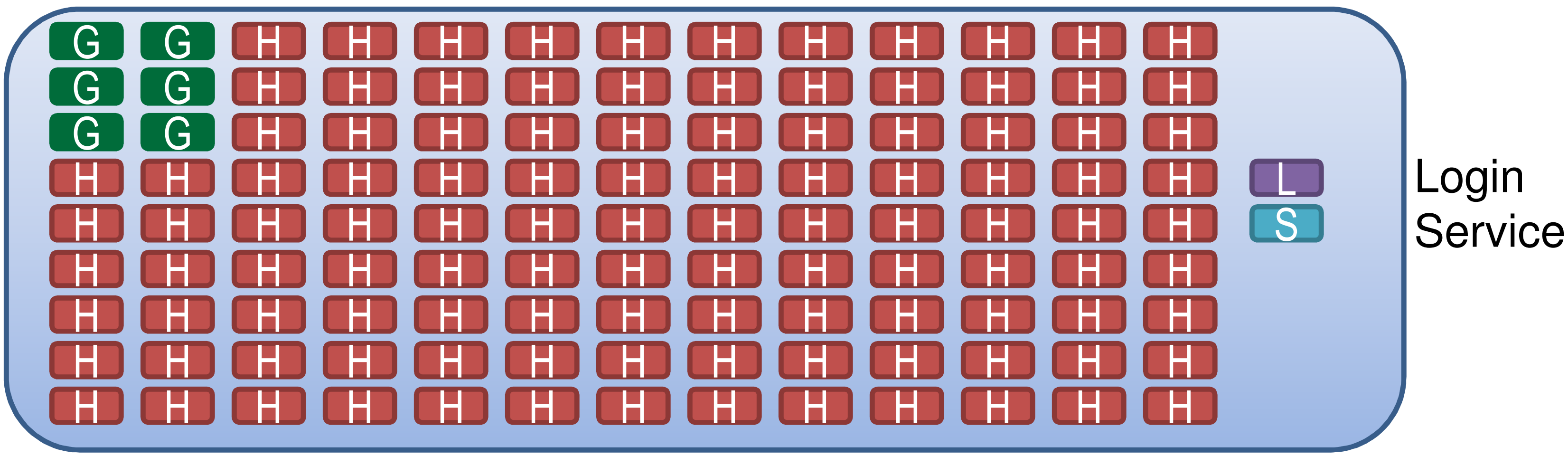
- A second major goal of modeling the power grid is to archive 1 year of data for reference and analysis
- This data will be used for:
  - Static and dynamic model validation
  - Generation of scenario library
    - Simulate frequency change of failures, incorporate into Event Detection.
- Requires high availability of data, task management for a large number of jobs
- These are offline processes but they "should never fail"

"Fat" Hadoop Node

- Hadoop is an open source framework for distributed processing of large data sets across clusters of computers.
- Focus on high-availability
  - Expect nodes to fail, react intelligently in software when they do
- Hadoop File System Built-In
  - Intelligently replicates data to guard against hardware failures
- Exposes map reduce interface to archived data
  - Useful for many statistical analyses

How Many Hadoop Nodes?

- 1 Year of Input Data: ~8.76PB
- GPU Compression: ~7.01 PB (but adds 2 GPU Nodes)
- Hadoop is designed to use commodity HDDs
  - 16*2 TB drives per node = 32 TB
  - Estimated Req: 220 Hadoop Nodes



Other Concern: Centralized vs. Distributed

- A suggested alternative is to distribute data locally at the sensors, rather than in a centralized repository
- This approach leads to several problems:
  - Most critically, additional latency in the 2-4 decision loop.
    - Example Latency Comparison (64 bytes)
      - ORNL to ORNL = 0.13 ms
      - ORNL to Georgia Tech = 14ms
      - ORNL to Stanford = 80ms
  - A centralized approach "pays" the 80ms once – then all data has arrived
  - Distributed approaches potentially pay this cost multiple times
  - Distributed approach lacks the infrastructure for batch analysis of historical data

Architecture Summary Table

Task	Requirement	Prototype Solution
Event Detection	Process 2TB/hr sensor data in real-time	GAEDA, 1.2 GB/s
Signature Search	Search all simulated scenarios in 2000ms	GAEDA, 1.5MM sig/s
Scenario Library	Exponential number of PG simulations	THYME on Keeneland, 58k simulations
Sensor Data Archive	Store 7.01 PB data	220 Node Hadoop Cluster

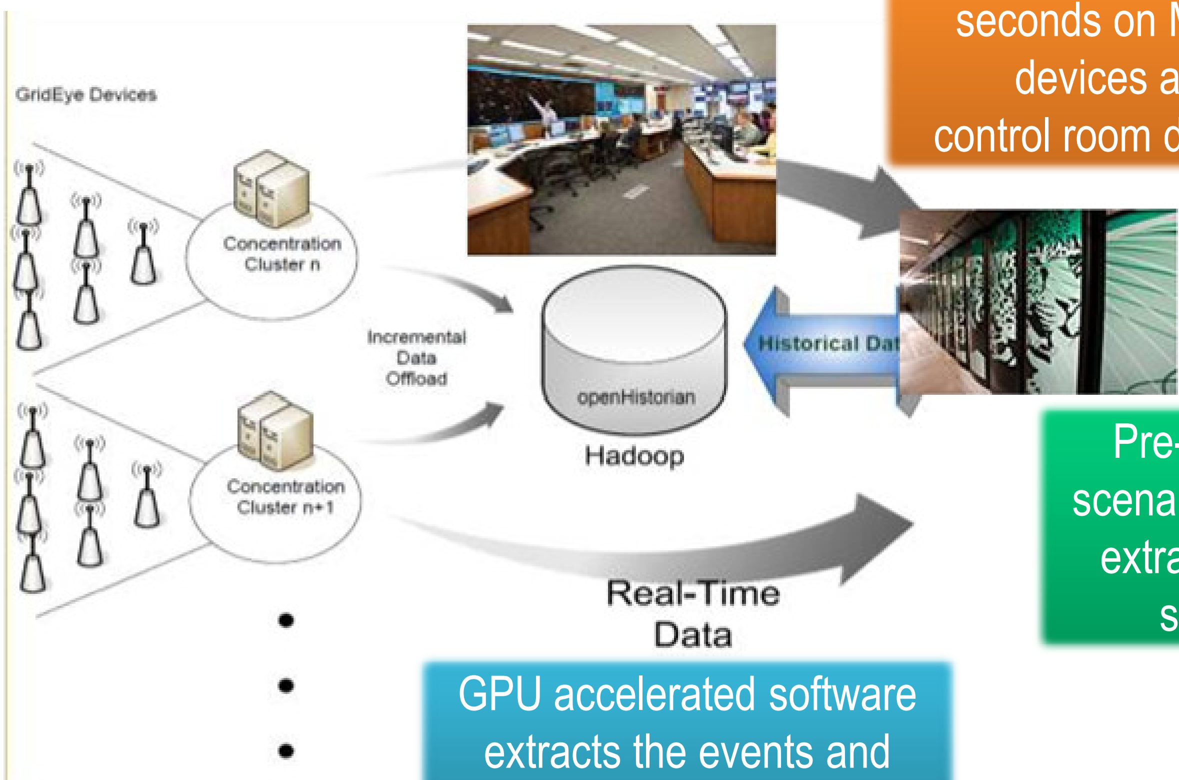
Conclusions

- Some methods that drive an end-to-end solution framework for monitoring the next generation of electric grid
- Encourage individual and shared situational awareness
- Permit coordinated emergency procedures for both areas of responsibility and observability
- Do not rely on knowing the specific modal frequencies in advance to design control scheme
- Account for spatial dependencies in sensor data

Thank You

Display to decision makers within 2-4 seconds on Mobile devices and control room displays

Detectors sense event within one second of event



Pre-calculated 10^8 scenarios with patterns extracted from HPC stored library

GPU accelerated software extracts the events and forwards to aggregator

Any Questions?

Femi Omitaomu, Ph.D.
omitaomuoa@ornl.gov